;; Auto-generated. Do not edit!

(when (boundp 'ur\_dashboard\_msgs::AddToLog)

(if (not (find-package "UR\_DASHBOARD\_MSGS"))

(make-package "UR\_DASHBOARD\_MSGS"))

(shadow 'AddToLog (find-package "UR\_DASHBOARD\_MSGS")))

(unless (find-package "UR\_DASHBOARD\_MSGS::ADDTOLOG")

(make-package "UR\_DASHBOARD\_MSGS::ADDTOLOG"))

(unless (find-package "UR\_DASHBOARD\_MSGS::ADDTOLOGREQUEST")

(make-package "UR\_DASHBOARD\_MSGS::ADDTOLOGREQUEST"))

(unless (find-package "UR\_DASHBOARD\_MSGS::ADDTOLOGRESPONSE")

(make-package "UR\_DASHBOARD\_MSGS::ADDTOLOGRESPONSE"))

(in-package "ROS")

(defclass ur\_dashboard\_msgs::AddToLogRequest

:super ros::object

:slots (\_message ))

(defmethod ur\_dashboard\_msgs::AddToLogRequest

(:init

(&key

((:message \_\_message) "")

)

(send-super :init)

(setq \_message (string \_\_message))

self)

(:message

(&optional \_\_message)

(if \_\_message (setq \_message \_\_message)) \_message)

(:serialization-length

()

(+

;; string \_message

4 (length \_message)

))

(:serialize

(&optional strm)

(let ((s (if strm strm

(make-string-output-stream (send self :serialization-length)))))

;; string \_message

(write-long (length \_message) s) (princ \_message s)

;;

(if (null strm) (get-output-stream-string s))))

(:deserialize

(buf &optional (ptr- 0))

;; string \_message

(let (n) (setq n (sys::peek buf ptr- :integer)) (incf ptr- 4) (setq \_message (subseq buf ptr- (+ ptr- n))) (incf ptr- n))

;;

self)

)

(defclass ur\_dashboard\_msgs::AddToLogResponse

:super ros::object

:slots (\_answer \_success ))

(defmethod ur\_dashboard\_msgs::AddToLogResponse

(:init

(&key

((:answer \_\_answer) "")

((:success \_\_success) nil)

)

(send-super :init)

(setq \_answer (string \_\_answer))

(setq \_success \_\_success)

self)

(:answer

(&optional \_\_answer)

(if \_\_answer (setq \_answer \_\_answer)) \_answer)

(:success

(&optional \_\_success)

(if \_\_success (setq \_success \_\_success)) \_success)

(:serialization-length

()

(+

;; string \_answer

4 (length \_answer)

;; bool \_success

1

))

(:serialize

(&optional strm)

(let ((s (if strm strm

(make-string-output-stream (send self :serialization-length)))))

;; string \_answer

(write-long (length \_answer) s) (princ \_answer s)

;; bool \_success

(if \_success (write-byte -1 s) (write-byte 0 s))

;;

(if (null strm) (get-output-stream-string s))))

(:deserialize

(buf &optional (ptr- 0))

;; string \_answer

(let (n) (setq n (sys::peek buf ptr- :integer)) (incf ptr- 4) (setq \_answer (subseq buf ptr- (+ ptr- n))) (incf ptr- n))

;; bool \_success

(setq \_success (not (= 0 (sys::peek buf ptr- :char)))) (incf ptr- 1)

;;

self)

)

(defclass ur\_dashboard\_msgs::AddToLog

:super ros::object

:slots ())

(setf (get ur\_dashboard\_msgs::AddToLog :md5sum-) "2054bf354d51f4d6a4dce330cf7bb805")

(setf (get ur\_dashboard\_msgs::AddToLog :datatype-) "ur\_dashboard\_msgs/AddToLog")

(setf (get ur\_dashboard\_msgs::AddToLog :request) ur\_dashboard\_msgs::AddToLogRequest)

(setf (get ur\_dashboard\_msgs::AddToLog :response) ur\_dashboard\_msgs::AddToLogResponse)

(defmethod ur\_dashboard\_msgs::AddToLogRequest

(:response () (instance ur\_dashboard\_msgs::AddToLogResponse :init)))

(setf (get ur\_dashboard\_msgs::AddToLogRequest :md5sum-) "2054bf354d51f4d6a4dce330cf7bb805")

(setf (get ur\_dashboard\_msgs::AddToLogRequest :datatype-) "ur\_dashboard\_msgs/AddToLogRequest")

(setf (get ur\_dashboard\_msgs::AddToLogRequest :definition-)

"string message

---

string answer

bool success

")

(setf (get ur\_dashboard\_msgs::AddToLogResponse :md5sum-) "2054bf354d51f4d6a4dce330cf7bb805")

(setf (get ur\_dashboard\_msgs::AddToLogResponse :datatype-) "ur\_dashboard\_msgs/AddToLogResponse")

(setf (get ur\_dashboard\_msgs::AddToLogResponse :definition-)

"string message

---

string answer

bool success

")

(provide :ur\_dashboard\_msgs/AddToLog "2054bf354d51f4d6a4dce330cf7bb805")